

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2017

Patrik Stloukal

Univerzita Pardubice

Fakulta elektrotechniky a informatiky

Návrh a implementace SW pro přenos vybraných informací na chytré hodinky

Patrik Stloukal

Bakalářská práce

2017

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2015/2016

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Patrik Stloukal**
Osobní číslo: **I12372**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Název tématu: **Návrh a implementace SW pro přenos vybraných informací na chytré hodinky**
Zadávací katedra: **Katedra informačních technologií**

Z á s a d y p r o v y p r a c o v á n í :

- Cílem práce je návrh a implementace aplikace pro náramkové hodinky Samsung Gear 2, která bude pomocí komunikačního rozhraní z již existující aplikace pro golfové hráče:
 - a) zobrazovat vybrané informace
 - b) nastavovat vybrané funkce
- V teoretické části práce bude proveden přehled v oblasti chytrých mobilních zařízení (smart-phonů) s důrazem na doplňková zařízení.
- V praktické části se student zaměří na vlastní návrh a implementaci SW řešení aplikace pro chytré hodinky.
- Výsledná aplikace bude otestována v reálném provozu.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

1.ULLMAN, L. PHP a MySQL. Brno, Computer Press Books, 2004

**2.MURPHY, Mark L. Android 2: Průvodce programováním mobilních aplikací.
Brno : COMPUTER PRESS, 2011. 376 s. EAN:9788025131947.**

Vedoucí bakalářské práce:

Ing. Jan Fikejz

Katedra softwarových technologií

Datum zadání bakalářské práce:

31. října 2015

Termín odevzdání bakalářské práce:

13. května 2016



prof. Ing. Simeon Karamazov, Dr.
děkan



L.S.



Mgr. Josef Horálek, Ph.D.
vedoucí katedry

V Pardubicích dne 31. března 2016

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 31. 08. 2017

podpis autora
Patrik Stloukal

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu práce panu Ing. Janu Fikejzovi, Ph.D. za cenné rady a možnost podílet se na vytváření zajímavé práce. Dále děkuji mamce a tátovi za podporu nejen při psaní této práce, ale i celého studia.

TITLE

Návrh a implementace software pro přenos vybraných informací na chytré hodinky.

ANOTACE

Tato bakalářská práce se zabývá vývojem software na chytré hodinky, jenž zajišťuje zobrazení základních informací a komunikaci s upraveným softwarem na mobilu.

Tato aplikace byla vytvořena v HTML 5 za pomoci CSS a JavaScriptu.

KLÍČOVÁ SLOVA

Singleton, HTML 5, CSS, Tizen, JavaScript

TITLE

Suggestion and implementation of software for the transmission of selected information on smart watches

ANNOTATION

This barchelor thesis is about development software on smart watches, which displays transmissioned information from mobile application. Mobile application was adjustment to sending and even receiving data.

This application was developed in HTML 5 and CSS and JavaScript.

KEYWORDS

Singleton, HTML5, CSS, Tizen, JavaScript

OBSAH

Úvod.....	12
1 Chytrý telefon	13
1.1 Mobilní aplikace.....	13
1.2 Budoucnost aplikací	13
1.3 Možnosti technologie	14
1.4 Android	14
1.4.1 Java	14
1.5 iOS.....	14
1.5.1 Objective-C.....	15
1.6 Windows 10 MOBILE	16
1.6.1 C #.....	16
1.7 Příslušenství	16
2 Použitá technologie.....	17
2.1 HTML5	17
2.2 CSS.....	17
2.3 JAVASCRIPT	17
2.4 Vývojová prostředí.....	18
2.5 Použitá zařízení	19
2.5.1 Samsung Galaxy S4	19
2.5.2 Samsung Galaxy Gear 2	19
3 Příprava.....	20
3.1 Instalace.....	20
3.1.1 Certifikát	20
3.2 Tutoriál.....	21
3.3 Seznámení s mobilní aplikací GolfGPS	22
3.4 Vývojová prostředí.....	23

3.5	Komunikace	24
3.6	Samsung Store.....	25
4	Návrh a implmemnetace	27
4.1	Struktura tizen webové aplikace	27
4.1.1	Běh aplikace.....	31
4.2	Úprava mobilní aplikace	31
4.2.1	Manifest	32
4.2.2	Singletony	32
4.2.3	Navázání spojení z pohledu aplikace na OS Android.....	33
4.2.4	Přijímání a odesílání dat	33
4.2.5	Odpojení.....	33
5	Instalační příručka.....	34
6	Závěr	35
7	Použitá literatura	36
8	Přílohy.....	38

SEZNAM ILUSTRACÍ A TABULEK

Obrázek 1 - Samsung Galaxy S4	19
Obrázek 2 - Mobilní aplikace v menu (vlevo) a na hřišti	23
Obrázek 3 - Zahájení připojení	25
Obrázek 4 - Ukázka seznamu chyb ze Samsung Store	26
Obrázek 5 - Struktura webové aplikace	27
Obrázek 6 - Ukázka listeneru při pohybu prstem po displeji	28
Obrázek 7 - Ukázka kódu pro vybrání CSS souboru	29
Obrázek 8 - Nastavení config.xml	30
Obrázek 9 - Ukázka Android Manifestu	32

SEZNAM ZKRATEK A ZNAČEK

CSS	Cascading Style Sheets
GPRS	General Packet Radio Service
GPS	Global Position Systém
GSM	Global System for Mobile Communications
HSPA	High Speed Packet Access
HTML	HyperText Markup Language
LTE	Long Term Evolution
MDA	Mobile Digital Assistant
PDA	Personal Digital Assistant
PHP	Hypertext Preprocessor
SAP	Samsung Accessory Protocol
SAS	Samsung Accessory Service
SDK	Software Development Kit
SMS	Short Message Service
WCDMA	Wideband Code Division Multiple Access
XML	Extensible Markup Language

ÚVOD

V současné době má člověk kolem sebe spoustu elektronických zařízení, ale hodinky nebo mobilní telefon má kolem sebe téměř nepřetržitě. Tyto zařízení, i se základními aplikacemi, nám ulehčují nejenom práci, ale i ostatní denní činnosti ve volném čase.

Stále vznikají novější a lepší zařízení, jako například chytré hodinky. Tato zařízení komunikují s mobilním telefonem a umožňuje k němu rychlý přístup, pro jednoduché nastavení nebo snadnější práci, jako volání, čtení emailu apod.

V teoretické části jsou popsány technologie spojeny s chytrými hodinkami a mobilním telefonem, a také vše potřebné k vytvoření companion¹ aplikace, tedy aplikace na chytré hodinky, která spolupracuje s aplikací spuštěnou na mobilním zařízení.

Praktická část se pak následně zaměřuje na vlastní návrh a implementaci aplikace pro chytré hodinky, která bude tvořit softwarovou nadstavbu nad již existující aplikací pro golfové hráče. Dále představí uživateli možný způsob aplikace podle zadaných požadavků. Zároveň uživatele seznámí s postupem a různými překážkami při implementaci Tizen aplikace. Obsahuje základní popis vytvořené aplikace, její strukturu, a hlavně komunikaci s upravenou Android aplikací.

¹ Tak se nazývá aplikace na chytré hodinky, která ke správnému chodu potřebuje druhou aplikaci na smartphonu.

1 CHYTRÝ TELEFON

Chytrý telefon, z angličtiny převzatý Smartphone, mobilní zařízení, ve kterém běží operační systém. Tyto zařízení se v posledních letech velice rozmohly.

První smartphone představila společnost IBM v roce 1993, který nabídl kromě telefonování i funkci kalendáře, rychlých poznámek, fax a e-mail. Následovala éra smartphonů např. od firem Nokia, Ericsson a Microsoft. Všechny měly zpočátku větší konstrukci s klávesnicí a mnohdy i bez dotykového displeje. V této době smartphony nebyly tak rozšířené. Až v roce 2001, kdy byl v Japonsku vytvořen smartphone s funkcí mobilního připojení k internetu. Ostatní smartphony nebyly tak úspěšné, protože byly zaměřené na práci, a ne na zábavu, a proto je používali jen obchodníci. Tohle bylo změněno až v roce 2007, kdy byl publikovaný první iPhone.

iPhone nabídl uživateli velký kapacitní dotykový display, chytré aplikace a celkově nové možnosti které nebyly do té doby běžné. Například audio konference, slušný fotoaparát, možnost kreslit a psát poznámky, pouštět si videa, připojit se k internetu, hrát hry.

Postupně na tuto situaci začali reagovat i další společnosti a následoval vývoj nezávislých mobilních operačních systémů, jako Android, FirefoxOS, Windows Phone a další. [13]

1.1 Mobilní aplikace

Mobilní aplikace je program nahráný do chytrého telefonu, který uživateli nabízí speciální služby. V dnešní době existují tisíce aplikací, které jsou volně ke stažení na oficiálních místech pro distribuci. U iOS je to App Store, u Android je Google play apod. Aplikace do různých systémů lze nahrát různými způsoby, jako přes rozhraní nebo přímo nahráním do paměťového úložiště. Aplikace jsou dostupné buď zadarmo s tím, že se v nich vyskytuje většinou reklama nebo je něčím omezena, pokud reklamu uživatel nechce, může si aplikaci koupit bez reklam, případně s více možnostmi a bez omezení.

Aplikace uživateli tedy přímo nabízí nějaké vlastnosti, které by mu měly něčím sloužit, pomáhat, vzdělávat a bavit ho.

1.2 Budoucnost aplikací

V budoucnosti chytrých hodinek se bude dbát na rozšíření možnosti hodinek a z pohodlnění každodenních činností. Dále je největším problémem ovládání chytrých hodinek. Dotykové displeje chytrých hodinek jsou zpravidla malé a uživatelům někdy přijde těžké například i správně vytvořit událost do kalendáře i s poznámkou. Některé chytré hodinky mají navíc i boční

tlačítka. Novější modely hodinek mohou být ovládány pomocí síly doteku, otáčení hodinek nebo naklonění hodinek. Například v aplikaci prohlížení mapy můžeme silou doteku po stranách displeje určit směr a rychlost posunu mapy. Při doteku a otočení hodinek kontrolujeme přiblížení nebo oddálení mapy a rychlost. Další ovládací prvek se spíše využije v jiných aplikacích. Pomocí doteku a naklonění hodinek se můžeme rychleji pohybovat v souborovém adresáři nebo můžeme nakreslit různé znaky, například při nakreslení znaku F se spustí Facebook atd. Díky těmto ovládacím prvkům se hodinky mohou změnit v ovladač a můžeme pomocí hodinek hrát hry.

1.3 Možnosti technologie

Chytré hodinky nám zjednodušují práci s mobilním telefonem. SMS zprávy přicházejí i na hodinky a máme i přístup na email. Můžeme přijímat hovory i na chytrých hodinkách, což nám umožňuje komunikaci i za volantem auta. Dále máme přístup k fotkám a hudbě. Chytré hodinky dokonce kontrolují životní styl. Počítají kroky, spálené kalorie a zpracovávají tato data do grafického záznamu.

1.4 Android

V roce 2007 firma Google začala vyvíjet Android jako open-source OS, aby měl každý přístup ke zdrojovému kódu Androidu a mohl ho měnit, za účelem vývinu aplikací a mobilních zařízení. V současné době je velmi rozšířen. OS Android je nahrán na více jak 24 000 zařízení a v Google Store je více jak 1 milión aplikací. [16]

1.4.1 Java

Java je vyšší programovací jazyk, který byl vyvíjen firmou Sun Microsystems od roku 1991. V roce 2009 Byla tato firma koupena společností Oracle Corporation, která vyvíjí tento jazyk do současnosti.

Je to objektově-orientovaný programovací jazyk, podobný C++, ale je zjednodušen. Byly zde odstraněny některé jazykové prvky, které způsobovaly běžné chyby. Kompilovaný Java kód lze spustit na většině počítačů pomocí Java Virtual Machine. JVM lze nainstalovat na většinu operačních systémů např. Unix, Windows. [17]

1.5 iOS

iOS je operační systém firmy Apple Computers, pojmenování má souviset s politikou pojmenování produktů, jako je iPhone, iPad, iPod, a iMac. Aktuální verzí je iOS 9.3, která nabízí spoustu inovací. Například funkce Night Shift, která dle geografické polohy zjistí, kdy u

vás zapadá slunce a zajistí, aby zařízení nevyzařovalo modré světlo, které komplikuje usínání. Ráno jej zase nastaví na normální barvy. Poznámky si můžete od této verze chránit heslem nebo otiskem prstu. Vylepšeny jsou také aplikace pro sledování zdraví a kondice, které jsou propojeny se zařízením Apple Watch.

System je navržen tak aby byl přehledný a snadno pochopitelný. Působí elegantně a snadno se ovládá. [14]

1.5.1 Objective-C

Programovací jazyk, který byl navržen Brad J. Cox napočátku 80. let 20. století. Tento jazyk vycházel z jazyka SmallTalk-80. Objective C je jedna z nadstavbových vrstev jazyka C která umožňuje tvorbu objektů a manipulaci s nimi. V roce 1988 byl jazyk licencován společností NEXT která byla v roce 1996 přebrána společností Apple Computer a ta vyvinula vývojové prostředí Cocoa. V roce 2007 byla společností Apple uvolněna aktualizace jazyka, která je označila jazyk jako Objective-C 2.0.

Po uveřejnění iPhone v roce 2007, vývojáři vyžadovali možnost vyvíjet aplikace pro toto zařízení. Z počátku firma Apple tuto možnost nevířala, ale později dala možnost vývojářům vyvíjet webové aplikace, což vývojáře moc neuspokojilo. Firma krátce na to oznámila, že uvolní prostředky pro vývoj takzvaných nativních aplikací.

V objective-C lze vyvíjet pro iOS, tak i pro Mac OS X a celkově se aplikace vyvíjejí na MacBook Pro. Brzy společnost nabídla vývojářskou sadu Software Development Kit (SDK) která obsahovala i simulátor iPhone, iPod Touch a iPad zařízení. V následujícím bloku kódu je k vidění ukázka kódu. [3]

```
#import <Foundation/Foundation.h>

int main(int argc, const char * argv[])
{
    NSAutoreleasePool * pool = [[NSAutoreleasePool alloc] init];
    NSLog(@"Bakalarska prace!");

    [pool drain];
    return 0;
}
```

1.6 Windows 10 MOBILE

Je mobilní operační systém vyvinutý firmou Microsoft a je následníkem systému Windows 8.1 PHONE. Tento systém je součástí spojení více platforem dohromady. Tedy Windows 10 budou mít všechny další zařízení, které produkuje Microsoft. Například Xbox, tablet Surface, chytré telefony a osobní počítače. [20]

1.6.1 C

V roce 1999 Anders Hejlsberg založil tým, který začal vývoj nového jazyka pro firmu Microsoft. Původně byl nazýván Cool, ale byl přejmenován na C#.

C# je založen na C++ a obsahuje podobné funkce jako Java. Myšlenka pro vývoj C# byla, aby tento programovací jazyk byl stejně rychlý jako C++, ale snadněji použitelný. Proto některé charakteristické funkce programovacího jazyka C++ nebyli přidány jako například vícenásobná dědičnost. I ukazatele byly značně zjednodušeny, protože jsou pro mnoho lidí náročné na použití.

Pro tvorbu C# aplikací se používá vývojové prostředí Visual Studio, které je také podporováno společností Microsoft. [18] [19]

1.7 Příslušenství

Nejznámější příslušenství k mobilu je asi handsfree, náramky a chytré hodinky. Handsfree plní jen jednu funkci. Po synchronizaci s telefonem je pouze na hovor. I handsfree v autě, kde je i displej na výběr z vašeho seznamu čísel, je handsfree vyrobené konkrétně na jednu funkci.

Chytré náramky se specializují na jedno odvětví, většinou na zdravotní stránku. Shrnují data o fyzické aktivitě uživatele, a dokonce i evidují průběh spánku. Uživatel tak má naprostý přehled o své fyzické činnosti. Tyto informace se posílají do mobilu, kdy mohou být dále zpracovávány. Takové náramky zpravidla mají i pár vedlejších funkcí. U některých modelů je dostupný budík, který je realizován pomocí vibračního motorku, nebo vás může upozornit na příchozí SMS.

Chytré hodinky mají funkce obou zmíněných příslušenství a ještě něco navíc. Příchozí a odchozí hovory či SMS i všechny funkce náramků. Krokoměr, měření srdečního tepu, kontrola spánku. Navíc k tomu je pár vychytávek. Zabudovaný fotoaparát přímo v hodinkách, ovladač na televizi nebo kalendář. V této široké nabídce funkcí si už uživatel snadno vybere oblíbené funkce. A pořád je to takový malý mobil, takže další aplikace, jako jednoduché hry nebo jízdni řády, se dají nainstalovat.

2 POUŽITÁ TECHNOLOGIE

V následující části bakalářské práce, jsou popsány technologie, které byly využity při vývoji aplikace.

2.1 HTML5

Tato specifikace značkovacího jazyka HTML5 byla uskutečněna v roce 2014 po celých 14 letech bez aktualizace. Přestože v tuto chvíli ještě HTML5 není dokončené, pro jeho výhody se rychle rozšířilo a prohlížeče začínají implementovat některé funkce HTML5. Je zde používána přímější struktura, takže se lépe programují, upravují a nachází chyby. HTML5 podporuje přehrávání videí a audio souborů. V dřívějších verzích jste museli spoléhat na plug-in některé společnosti. Tím, že se musely stáhnout a taky pořád updatovat, se vytvořilo bezpečnostní riziko. Tato verze již podporuje offline webové aplikace. V manifestu aplikace zjistí, které soubory potřebuje pro danou aplikaci, stáhne do počítače a udržuje je aktuální. [4]

2.2 CSS

CSS je stylovací jazyk, který slouží k definování vzhledu HTML dokumentu. Rozdíl mezi HTML a CSS je, že HTML slouží k definici struktury obsahu a CSS se používá k formátování tohoto obsahu. [2]

2.3 JAVASCRIPT

Znám jako jeden z nejvíce nepochopených jazyků na světě. Pod svou zdánlivou jednoduchostí však skrývá řadu možností a je dnes nasazený ve velkém množství vysoce profesionálních webových řešení. Hlubší znalost tohoto nástroje je dnes nezbytná jak pro vývojáře mobilních aplikací, tak hlavně, a především pro vývojáře aplikací webových.

Prvopočátky tohoto jazyka sahají až do poloviny devadesátých let, konkrétně do roku 1995, kdy byl vytvořen pro potřeby prohlížeče Netscape a následně vydán s druhou verzí Netscape v roce 1996. Původně byl Javascript nazván LiveScript, k přejmenování na dnešní název došlo pro účely marketingu a mělo být krokem k využití popularity ekosystému JAVA. Výsledkem tohoto rozhodnutí je zmatek, který přetrvává prakticky do dnes.

O několik měsíců později Microsoft vydal implementaci JScriptu, který byl s JavaScriptem kompatibilní a vydal jej v prohlížeči Internet Explorer 3. Během následujících měsíců byl Netscapem navržen JavaScript ke standardizaci komisí Ecma čímž dal vzniknout první edici

ECMAScriptu. Výrazných vylepšení se dočkal ECMAScript v edici 3 ještě v roce 1999 od které je již poměrně stabilní. Dále byla vyvíjena ještě čtvrtá edice, která ovšem přijata nebyla, ale byla od ní odvozena edice pátá vydána roku 2009 a šestá vydaná v půli roku 2015.

Na rozdíl od většiny ostatních jazyků, Javascript sám o sobě neimplementuje vstupní a výstupní operace. Na místo toho je navržen tak, že běží v prostředí, které těmito prostředky samo disponuje a Javascript je pouze využívá ke komunikaci s okolním světem. Vůbec nejběžnějším prostředím, kde lze Javascript spouštět je samozřejmě webový prohlížeč, existují však i jiné implementace interpretů Javascriptu, které můžeme najít v produktech Adobe, Yahoo Widget enginu anebo dokonce na serverových řešení, jako jsou Node.js, NoSQL databázové systémy, desktopová grafická prostředí anebo pro nás nejdůležitější vestavěné systémy. [1]

2.4 Vývojová prostředí

Pro vývoj aplikací byla použita oficiální vývojová prostředí, a to Android Studio a Tizen SDK.

Tizen SDK

Tizen je vývojové prostředí dodané asociací Tizen, která se skládá například z firem Fujitsu, Huawei, Intel, Samsung. Instalace je jednoduchá. Stačí stáhnout a nainstalovat balíček Tizen SDK. Je vhodné stáhnout a dodatečně nainstalovat balíček pro vývoj aplikace na hodinky nebo mobily. [12]

Android Studio

Vývojové prostředí založené na IntelliJ IDEA a bylo představené roku 2013. Instalace je velmi jednoduchá, časově nenáročná, stačí nainstalovat instalační balíček, který je zdarma ke stažení a samozřejmě JDK. Android Studio je přehledné. Možnost uspořádání oken. Program funguje rychle a svižně, ale je náročný na operační paměť. Je zde možná emulace android systému. Program je pravidelně aktualizován, takže je možný vývoj aplikace pro aktuální verzi. [10] [11]

2.5 Použitá zařízení

Aplikace nebyla vyvíjena pouze na virtuálním stroji, ale také na fyzickém zařízení, na kterém byla testována.

2.5.1 Samsung Galaxy S4

Mobilní telefon Galaxy S4 je vlajkovou lodí jihokorejské společnosti Samsung. Každý rok společnost prezentuje nový model této řady, tedy po tomto telefonu následuje S5, S6, S7. Telefon má 5" palcový AMOLED displej s rozlišením 1080 x1920 pixelů (441 PPI). Nabízí možnost pracovat v síti LTE, GSM, HSPA, EDGE, WCDMA a GPRS. Operační paměť má 2048 MB a procesorem Qualcomm Sharpdragon 600 s frekvencí 1 900MHz. Mobilní telefon má předinstalovaný operační systém Google Android 4.2 (Jelly Bean). Na obrázku 1 je ilustrace použitého zařízení Samsung Galaxy S4, který byl vybrán z důvodů dostupnosti a také Samsung Galaxy Gear 2 lze spárovat jen s mobily od firmy Samsung.



Obrázek 1 - Samsung Galaxy S4

Zdroj: <http://www.mobilmania.cz/katalog-mobilu/samsung-galaxy-s4/sc-63-ci-48098-c-1>

2.5.2 Samsung Galaxy Gear 2

Druhá generace chytrých hodinek od společnosti Samsung. Oproti první generaci již mají operační systém Tizen. Hodinky mají 1,63" palcový AMOLED displej a dvoujádrový procesor s frekvencí 1GHz. Tato generace chytrých hodinek není kompatibilní s telefony jiných značek.

3 PŘÍPRAVA

K vývoji aplikace je potřeba nejen software, ale i materiály či příručky se stejným tématem. Se správným vedením lze ušetřit spoustu času. Z tohoto důvodu jsem napsal pár připomínek, které mě stály cenný čas.

3.1 Instalace

Jako první krok je potřeba nainstalování vývojových prostředí. Android studio se instaluje jednoduše. Stačí stažení jednoho souboru, ze kterého je poté nainstalováno vývojové prostředí. Poté je potřeba stáhnout ze stránek Samsungu Accessory SDK a přidat knihovny do projektu. Tyto knihovny podporují komunikaci se Samsung hodinkami.

Zatímco vývojové prostředí Tizen SDK, které je potřeba na vytvoření aplikace na hodinky, je složitější. Stažení instalátoru a instalačního balíčku je jednoduché. Ale je potřeba i dvě rozšíření, rozšíření vývojového prostředí pro hodinky a rozšíření pro podporu certifikátů. Na stránkách Samsungu je záložka Extensions (Rozšíření), ale zde je pouze návod, jak tyto rozšíření nainstalovat. Samotné soubory se nacházejí v technické dokumentaci stránek Samsung.

Další krok, instalace, je již snadný a lze pokračovat podle návodu. Jako poslední je potřeba nainstalovat ovladače k připojení hodinek a mobilu do vývojových prostředí.

Přestože stránky Samsungu jsou reprezentativní, velice dobře členěné s vynikajícím designem a spoustou informací, může být překvapující například chybějící odkaz na soubor na stránce Rozšíření, která je, podle způsobu obsahu, zaměřená na začínající uživatele s počítačem. Podle mého názoru společnost Samsung je zaměřená na vývoj chytrých hodinek. Je to novější technologie a vyvíjí se s velkou rychlostí. Ale, i když si zakládají na podpoře svých výrobků, základní materiály a výborné tutoriály vyžadují čas.

3.1.1 Certifikát

Pro správné fungování aplikace je nutné získat a dva certifikáty. Prvním je autorský certifikát, který se přiloží k aplikaci. Až je aplikace nahraná v Samsung Store, tak update této aplikace lze pouze se stejným autorským certifikátem.

Druhý certifikát je certifikát zařízení. Ten autorizuje pouze ty hodinky, které byly v době žádosti připojené k počítači, k instalování aplikací jinak než přes Samsung Store, za účelem testování aplikace.

Při vývoji samostatné aplikace není nutné mít fyzické zařízení hodinek. Vývojové prostředí Tizen SDK obsahuje emulátor hodinek a lze zde nastavit i rozlišení a tvar hodinek. Ale cílem této bakalářské práce jsou dvě aplikace, které mezi sebou komunikují, tak je zde potřeba i fyzické zařízení na otestování komunikace. V takovém případě je i certifikát zařízení potřebný.

Certifikáty lze získat přímo ve vývojovém prostředí Tizen.

Nejdříve se vygeneruje soubor CSR, kde se zadají základní informace o autorovi, hesla pro certifikáty a pokud chceme i certifikát zařízení je potřeba mít připojené hodinky, protože tento soubor obsahuje i informace o hodinkách.

V dalším kroku jsme přeměrováni na stránky Samsungu pro vývojáře. Zde je potřeba založení účtu a CSR soubor. Certifikáty přichází po dvou dnech na e-mail.

Certifikáty se vloží do vývojového prostředí a certifikát zařízení se musí nahrát na hodinky, jinak nám bude nahlášena chyba při instalaci aplikace. Certifikát zařízení se nahrává v Tizen SDK v možnostech konkrétních hodinek volbou *Permit to install applications*.

Později byl tento systém certifikátů přepracován. Postup žádosti je stejný, ale certifikáty nejsou posílány e-mailem, ale ihned poslány zpět do vývojového prostředí, takže celý proces je zkrácen na pár minut.

3.2 Tutoriál

Na stránkách Samsungu, v sekci pro vývojáře, jsou tutoriály, a i několik vzorových aplikací ve stavu projektových složkách. Aplikace může být nativní, za použití jazyka C, anebo webová, za použití HTML, CSS a JavaScriptu. [5]

Původně byl vybrán nativní způsob, protože je lepší v kontrole komunikace s mobilem. Po zhlédnutí tutoriálu a vyzkoušení vzorové aplikace, tato aplikace nelze spustit. Byla mě nahlášena chyba, že nejsou připojené hodinky. Řešení této chyby, jak na různých fórech nebo i na stránkách a fóru Samsungu, jsou jednoduché. Připojit hodinky bezdrátově k laptopu. Zařízení Samsung Gear 2 nepodporují připojení přes Wi-Fi. Při dalším hledání řešení problému bylo zjištěno, že Gear 2 nepodporují nativní aplikace.

Stránky Samsungu byly v době tvorby bakalářské práce hodně často předělávané a jejich poslední aktualizace byla značně ve spěchu. Stránky byly aktualizované na dřívější nejnovější model Samsung Gear S2. Nebylo uvedeno, že některé části kódu nebo celý nativní přístup není v dřívějších verzích hodinek podporován.

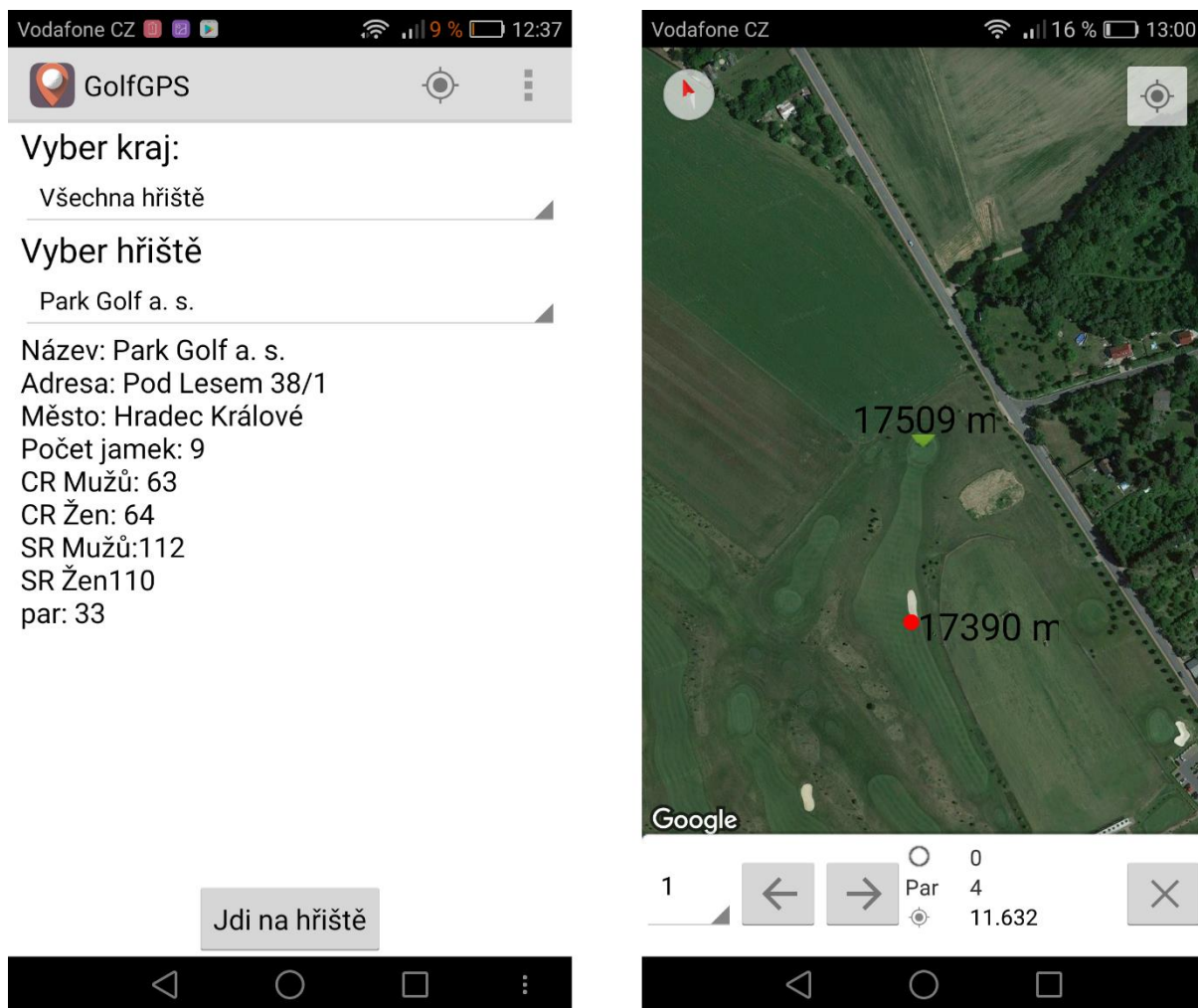
Tutoriál webové aplikace byl nepřehledný a v některých částech byla nutná dobrá znalost Tizen Web API, bez které výsledná aplikace nebyla kompletní. Tento tutoriál byl pro mě především na vysvětlení, jak webová aplikace na hodinkách funguje, zvláště připojování k druhé companion aplikaci. Nejen Samsung, ale i Tizen má na svých stránkách různé příručky o jejich rozhraní, jak začít s vývojem aplikace a i o companion aplikacích a podobně. Jsou zde i ukázkové aplikací, jak samostatné, tak i companion. Takže i když tutoriál nebyl úplně jednoznačný, je zde spousta materiálů na dovysvětlení a pochopení problematiky, a také lze najít pár neoficiálních stránek řešící právě základy těchto aplikací.

3.3 Seznámení s mobilní aplikací GolfGPS

Aplikace GolfGPS vznikla jako bakalářská práce pana Bc. Hamala s cílem usnadnit golfovým hráčům lepší orientaci ve vzdálenosti mezi aktuální polohou hráče a hranou jamkou [15].

Nejsem tedy autorem aplikace GolfGPS, ale pro tuto bakalářskou práci nám bude stačit jen lehké seznámení a úprava malé části aplikace.

Na obrázku č. 2 lze vidět menu aplikace, kde je na výběr ze 3 golfových hřišť. Po výběru hřiště zmáčkneme tlačítko v dolní polovině displeje *Jdi na hřiště*. Tato část aplikace je pro nás nejdůležitější. Je zde za pomoci Google Map znázorněno hřiště. Jamka a odpaliště míčku je popsáno vzdáleností k mobilu. V liště u dolní části displeje jsou údaje, jako číslo jamky, par a přesnost GPS. Tyto údaje se vzdáleností jamky od mobilu jsou pro naši práci stěžejní. Já jsem zvolil snadné řešení. Najít ve zdrojovém kódu místa, kde jsou tato čísla vypsána a aktualizována a poslat je i na hodinky.



Obrázek 2 - Mobilní aplikace v menu (vlevo) a na hřišti

Zdroj [15]

3.4 Vývojová prostředí

Tizen SDK je rozšíření pro Eclipse Framework, je přehledný a práce bude probíhat v pořádku, jen je třeba vyhnout se následujících chyb.

Do balíčku aplikace se nepřiloží všechny soubory. Například po změně několika řádků v kaskádových stylech si chceme ověřit změnu, ale po spuštění aplikace se na hodinkách objeví chybová stránka se stavovým kódem 404 a index.html není nalezen. Další spuštění této aplikace, i po restartování systémů, je se stejným výsledkem. I když zvlášť vytvoříme balíček aplikace před spuštěním aplikace, není tato chyba vyřešena. Opravení této chyby je lehké. Jestli jsme spustili aplikaci v debug módu, tak stačí aplikaci spustit v normálním módu, a naopak po spuštění v normálním módu spustíme aplikaci v debug módu.

Další chyba je podobná té první. Ale po spuštění aplikace nejsou zjevné žádné změny. Balíček aplikace se vůbec nevytvořil a ani při opětovném spuštění se balíček nevytvoří. Řešení této chyby je stejné jako u první chyby.

Poslední chyba nastává při spouštění aplikace v režimu debug, kde je možnost, že se instalační balíček špatně zkompiluje, ale při opětovném spuštění je aplikace spuštěna v pořádku, za předpokladu, že se nevyskytne žádná z předchozích chyb.

S vydáním Samsung Gear S3 bylo vytvořeno nové vývojové prostředí Tizen Studio. Bylo to vydáno až ke konci mé bakalářské práci, takže nebylo moc testováno, ale při vyzkoušení byly tyto chyby odstraněny.

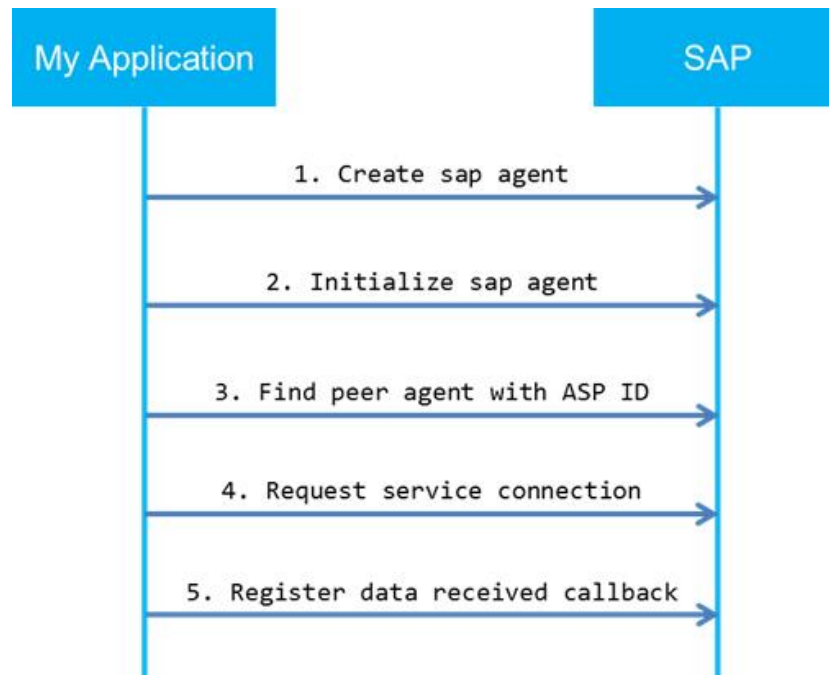
V režimu debug je možnost vizuální úpravy aplikace. Při zvolení jednotlivých tagů se ukáže seznam všech kaskádových stylů, které jsou uplatněny na vybraný tag, a příslušný tag je zobrazen ve spuštěné aplikaci.

Přestože jsem dříve neznal Android Studio, tak práce v tomhle vývojovém prostředí probíhala v pořádku a nenalezl jsem zde žádné větší komplikace.

3.5 Komunikace

Aplikace je typu companion, tedy aplikace na hodinkách spolupracuje s aplikací na mobilu. Zvolil jsem Gear aplikaci jako consumer a mobilní aplikaci jako provider, protože v tomhle případě hodinky zahajují připojení aplikací. Mobilní aplikace funguje i samostatně a ke svému běhu nepotřebuje hodinky, tedy nemusí zbytečně inicializovat spojení.

Na obrázku č. 3 lze vidět postup navazování spojení. Hodinky nejprve vytváří *SAP agenta*, který obsahuje informace o spojení jako např. jméno aplikace a identifikátor. Tento agent je poslán broadcastem. Mobilní aplikace si vytvoří svého *SAP agenta* a porovná, jestli jsou stejné informace o spojení a odpoví. Když aplikaci na hodinkách přijde kladná odpověď, tak pošle požadavek o spojení a čeká na odpověď od mobilní aplikace. Při navázání spojení vznikne *socket*, který se používá na odesílání a přijímání dat.



Obrázek 3 - Zahájení připojení

Zdroj: <http://developer.samsung.com/gear/develop/creating-your-first-app/web-companion/interact-device>

3.6 Samsung Store

Kromě testování aplikací, jsou možné instalace aplikací jen přes mobilní aplikaci Gear Manager. Gear Manager ale instaluje pouze ty aplikace, které jsou zaregistrované a prodávané v obchodě Samsungu.

Účet i přidání aplikace do obchodu je zdarma a proces je jednoduchý. Je zde možnost přidání aplikace do testovacího obchodu, kde je možné testovat aplikaci zejména instalace a její chod. Při požadavku přidání aplikace do standardního obchodu je aplikace testována zaměstnanci Samsungu. Když aplikace nesplňuje požadavky obchodu, je zpět poslán seznam chyb a video testování aplikace. Seznam chyb a link na video lze vidět na obrázku 4.

Validation Result Report

Samsung Galaxy Apps

Copyright © 2015 Samsung Electronics Co., Ltd. All rights reserved.

► Issue must be fixed

Issue Type	Title	Description	Detected Device Group	Attached File
Ad-Hoc	There is no deep link for app to download companion app on Host device	<p><Defect> There is no deep link for app to download companion app on Host device</p> <p><Procedure> 1.Installed app 2.Apply the watch face > Check on Gear whether clock has link to download companion app or not</p> <p><Expected Result> After applying the watch face, a pop-up (deep link) that requires to download companion app on phone should be shown up</p>	Gear S3 - Premium Galaxy (FHD WQHD), Gear S3 - Premium Galaxy (HD qHD), Gear S2 - Premium Galaxy (HD qHD), Gear S2 - Premium Galaxy (FHD WQHD)	1_Function Error_GPS.mp4

► Issue recommended to be fixed

Issue Type	Title	Description	Detected Device Group	Attached File
Phone comp	Gear - Premium Galaxy(WVGA) device groups are only on sale in China.	<p><Defect> Gear devices, which are tied with Premium Galaxy(WVGA), only supported in China is selected as sales device, but China is not selected as sales country.</p> <p><Prodedure> Check if seller selects China as sales country when 'Gear - Premium Galaxy(WVGA)' device groups are selected.</p> <p>*Gear BT only - Premium Galaxy (WVGA) *Gear Standalone - Premium Galaxy (WVGA) *Android Gear BT only - Premium Galaxy (WVGA) *Gear BT only - Premium Galaxy (WVGA) *Gear Standalone - Premium Galaxy (WVGA)</p> <p><Expected Result></p>	Common	-

Obrázek 4 - Ukázka seznamu chyb ze Samsung Store

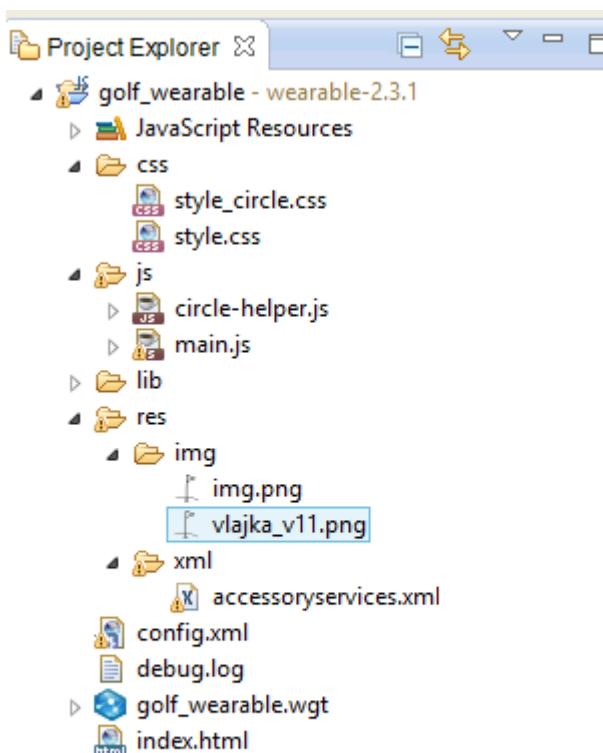
Zdroj: Vlastní

4 NÁVRH A IMPLMEMNETACE

Jedním z hlavních cílů bakalářské práce bylo vytvořit aplikaci na chytré hodinky a upravit stávající aplikaci na mobilním zařízení. Nejprve se seznámíme s Gear aplikací, její strukturou a důležitými částmi aplikace. Následuje Android aplikace, kde si ukážeme, které části aplikace se musí upravit a dojde k vysvětlení připojování z pohledu mobilní aplikace.

4.1 Struktura tizen webové aplikace

Aplikace se rozděluje na dvě části. HTML a kaskádové styly vytváří grafické znázornění aplikace a druhá část aplikace, která se skládá z JavaScript a XML souborů, uskutečňuje běh aplikace. Obrázek č. 5 vyobrazuje výslednou strukturu Gear aplikace.



Obrázek 5 - Struktura webové aplikace

Zdroj: Vlastní

Kaskádové styly

Prizpůsobují zobrazení jednotlivých elementů na stránce HTML a vytváří tak celkové rozložení aplikace. Jsou zde dva soubory, když je displej hodinek čtvercový použije se soubor `style.css`, a když je naopak kulatý, tak se použije soubor `style_circle.css`.

JavaScript

Soubory v této složce zajišťují chod aplikace. `Circle-helper` je podpůrný soubor vytvořený založením projektu. Soubor `main` zahajuje spojení s druhou aplikací, udržuje komunikaci, příchozí proud dat rozděluje do proměnných, která jsou následně vypsaná na displeji. Na obrázku 6 je zobrazen listener na událost `swipe`, neboli při pohybu prstu po obrazovce doleva nebo doprava je poslána zpráva druhé aplikaci o směru pohybu, aby došlo k posunu jamky.

```
var element = document.getElementById("main");
tau.event.enableGesture(element, new tau.event.gesture.Swipe(
{
  orientation: "horizontal"
}));
element.addEventListener("swipe", function(e)
{
  if(e.detail.direction=="left"){send(2);}
  if(e.detail.direction=="right"){send(1);}
});
```

Obrázek 6 - Ukázka listeneru při pohybu prstem po displeji

Zdroj: Vlastní

Zdroje aplikace

Složka `res` neboli `resources` obsahuje XML soubory a obrázky. Dále je rozdělena do podadresářů.

Obrázky

Složka `img` obsahuje dva obrázky. I když jsou si velmi podobné, jeden je upraven. `Img.png` slouží estetickým účelům aplikace. Druhý obrázek je upraven, protože je použit jako ikona aplikace, kterou reprezentuje jak v menu hodinek, tak i v Samsung Store.

XML

V této aplikaci je použit jen jeden přídatný XML soubor. `Accessorieservices` je soubor, který obsahuje nastavení o komunikaci s mobilní aplikací. Tyto informace jsou velmi důležité při žádání o spojení, kdy je kontrolováno jméno a id aplikace. Dále je zde nastavena komunikace přes bluetooth a role jako consumer.

HTML

Soubor HTML s názvem `index`, pro konvenční název. Pomocí značek je nastíněna grafická podoba aplikace. Je připojen JavaScript soubor a podle tvaru hodinek i správný CSS soubor. Jednu z možných implementací lze vidět na obrázku č. 7.

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width,user-scalable=no" />
<meta charset="utf-8">
<title>Golf</title>
<link rel="stylesheet" href="lib/tau/wearable/theme/default/tau.css">
<link rel="stylesheet" media="all and (-tizen-geometric-shape: circle)"
      href="lib/tau/wearable/theme/default/tau.circle.css">
<link rel="stylesheet" media="all and (-tizen-geometric-shape: circle)"
      href="css/style_circle.css">
<link rel="stylesheet" media="all and (-tizen-geometric-shape: rectangle)"
      href="css/style.css">
<!-- load theme file for your application -->

</head>
```

Obrázek 7 - Ukázka kódu pro vybrání CSS souboru

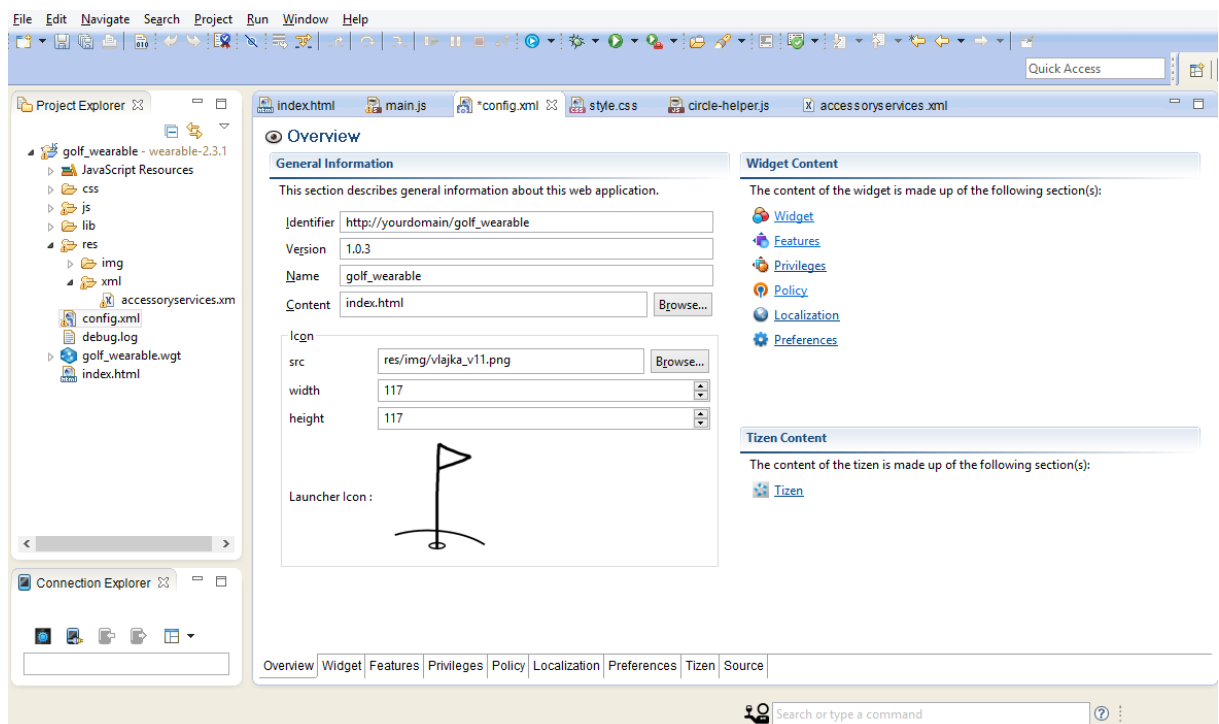
Zdroj: Vlastní

Nastavení `config.xml`

Tento soubor obsahuje informace např. jaký displej aplikace podporuje nebo odkaz na mobilní aplikaci, proto je důležitý při nahrávání aplikace do Samsung Store. Soubor `config` má 9 záložek.

- Overview – V této záložce se nastavuje jméno aplikace, její verze. Dále se zde vkládá ikona aplikace. Tahle záložka je pro představu vyobrazena na obrázku č. 8.
- Widget – Obsahuje informace o autorovi, tato záložka je nepovinná.

- Features – Softwarové nebo hardwarové požadavky na fyzické zařízení. Obvykle rozlišení displeje nebo tvar. My zde máme i požadavek na accessory protocol, to nám zajistí, že hodinky podporují právě komunikaci s mobilem.
- Privileges – Zde se mohou specifikovat, které API je zpřístupněno a používáno webovou aplikací. V naší aplikaci je specifikováno API accessory protocol.
- Policy – Nastavení přístupu k externím webovým zdrojům.
- Localization – Překlad názvu a popisu aplikace podle místního jazyka.
- Preferences – Zde lze nastavit proměnné ve formě jméno-hodnota, které mohou být zpřístupněné pomocí Widget Interface API.
- Tizen – Tato záložka obsahuje specifické informace o tizen aplikaci. Mimo ID aplikace je zde pro nás důležitá část metadata. V této sekci nastavíme adresu mobilní aplikace a přiložíme accessorieservices XML.
- Source – Nastavení ve všech předchozích záložek se ukládá se správnou syntaxí do XML souboru. V této záložce je otevřen tento soubor a je možné sem předchozí nastavení přímo napsat či upravit.



Obrázek 8 - Nastavení config.xml

Zdroj: Vlastní

4.1.1 Běh aplikace

Po spuštění aplikace a aplikování kaskádových stylů se zavolá funkce `window.onload`, která přidá listener pro zavření aplikace při pohybu prstem z horního okraje obrazovky směrem dolů a také volá funkci `connect` každé 3 vteřiny, dokud není aplikace připojená k druhé companion aplikaci. Připojování a posílání dat je znázorněno v příloze A jako diagram.

Spojení

Funkce `connect` zavolá funkci `webapis.sa.requestSAAgent` pro získání specifického *SAAgent* z accessory service profilu.

Dále je volána funkce `SAAgent.findPeerAgents`, která hledá zařízení se stejným rozhraním. Funkce pošle *SAP agenta* broadcastem a listener čeká na příchozí odpověď. Funkce `peerAgentFindCallback` reaguje na odpověď, porovnává vytvořeného *SAP agenta* s příchozím *SAP agentem*. Se shodou jména aplikace a ID je vytvořen listener spojení (`SAAgent.setServiceConnectionListener`) a poslána žádost o připojení. Aplikace čeká na odpověď. Pokud je žádost přijmuta, tak přijde aplikaci instance objektu *SASocket*.

Posílání a přijímání dat

K přijímání dat je použita funkce `onreceive` s parametry `channelId` a `data`. `channelId` je číslo, které musí být stejné jako číslo `id` v `accessoryservices.xml` a parametr `data` je proud dat poslán z druhé aplikace. V této funkci je proud dat parsován do proměnných a jsou aktualizována na displeji.

Posílání dat je zpřístupněno funkcí `socketu sendData`. Parametry jsou opět `id` kanálu a zpráva.

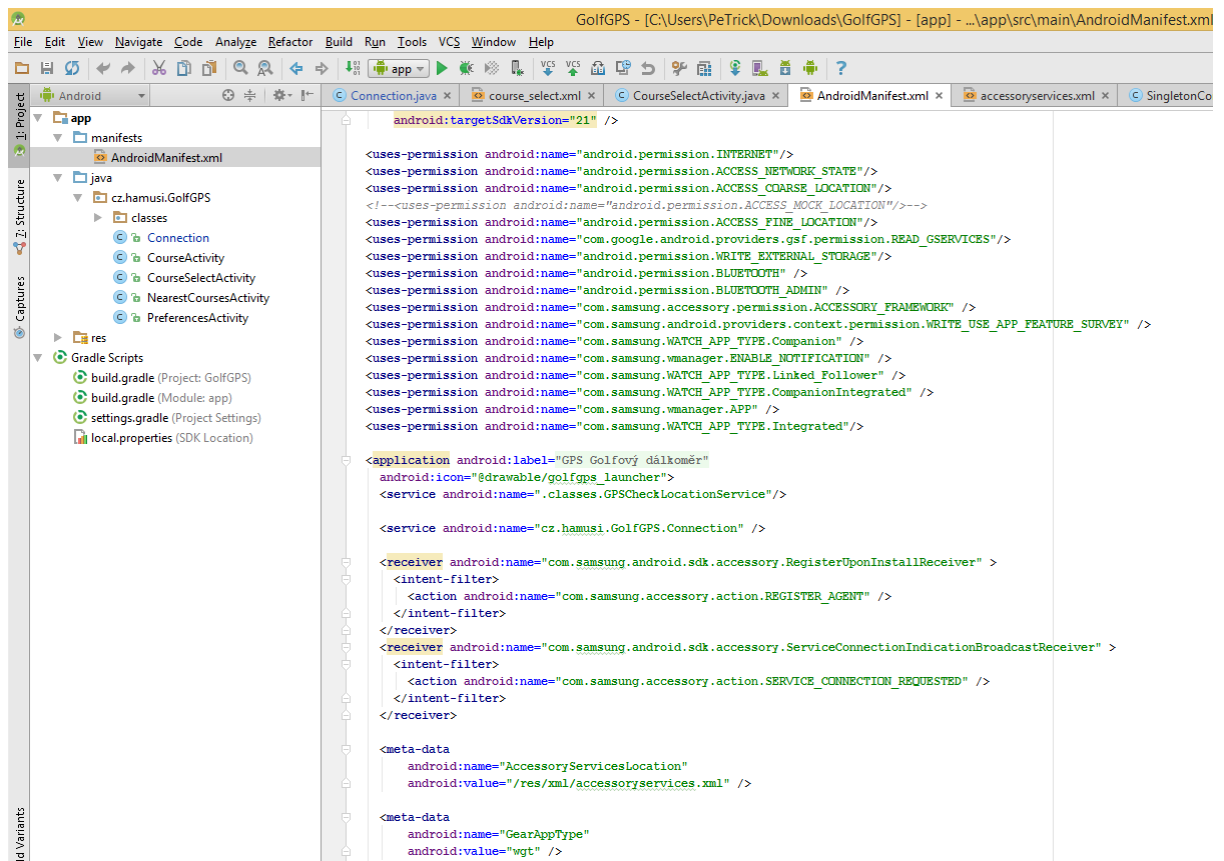
Posílání dat je použito v naší aplikaci spolu s událostí `swipe`, tedy při pohybu prstem po obrazovce v horizontálním směru doleva nebo doprava, je posláno číslo 1 nebo 2. Směr vpravo je znázorněn číslem 1 a levý směr číslem 2.

4.2 Úprava mobilní aplikace

Aplikaci stačí upravit na čtyřech místech. Musí se upravit Manifest aplikace. Přidat třídu `Connection`, která bude komunikovat s hodinkovou aplikací. Upravit aktivitu hřiště. Tam kde se informace vypisují na displej mobilu, je poslat i na hodinky. A v posledním kroku, aby aktivita hřiště měla přístup k instanci třídy `Connection` a naopak

4.2.1 Manifest

Jedná se o nejdůležitější soubor v celé struktuře aplikace, vytváří se při založení projektu. Obsahuje oprávnění a aktivity aplikace. Obrázek č. 9 ukazuje přidání oprávnění pro použití bluetooth a různé další povolení pro hodinky. Do manifestu je přidán i XML `accessoryservices`, který je téměř stejný jako v aplikaci na hodinkách. Největší změna je u role připojení, zde je nastaven provider. Jak lze na obr. 6 vidět, zde deklarujeme broadcast přijímače, aby broadcast zprávy byly vyřizovány *SAS frameworkem*.



```
android:targetSdkVersion="21" />

<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<!--uses-permission android:name="android.permission.ACCESS_MOCK_LOCATION"/>-->
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="com.samsung.accessory.permission.ACCESSORY_FRAMEWORK" />
<uses-permission android:name="com.samsung.android.providers.context.permission.WRITE_USE_APP_FEATURE_SURVEY" />
<uses-permission android:name="com.samsung.WATCH_APP_TYPE.Companion" />
<uses-permission android:name="com.samsung.wmanager.ENABLE_NOTIFICATION" />
<uses-permission android:name="com.samsung.WATCH_APP_TYPE.Linked_Follower" />
<uses-permission android:name="com.samsung.WATCH_APP_TYPE.CompanionIntegrated" />
<uses-permission android:name="com.samsung.wmanager.APP" />
<uses-permission android:name="com.samsung.WATCH_APP_TYPE.Integrated" />

<application android:label="GPS Golfový dálkoměr"
    android:icon="@drawable/golfgps_launcher"
    <service android:name=".classes.GPSCheckLocationService"/>

    <service android:name="cz.hamusi.GolFGPS.Connection" />

    <receiver android:name="com.samsung.android.sdk.accessory.RegisterUponInstallReceiver" >
        <intent-filter>
            <action android:name="com.samsung.accessory.action.REGISTER_AGENT" />
        </intent-filter>
    </receiver>
    <receiver android:name="com.samsung.android.sdk.accessory.ServiceConnectionIndicationBroadcastReceiver" >
        <intent-filter>
            <action android:name="com.samsung.accessory.action.SERVICE_CONNECTION_REQUESTED" />
        </intent-filter>
    </receiver>

    <meta-data
        android:name="AccessoryServicesLocation"
        android:value="/res/xml/accessoryservices.xml" />

    <meta-data
        android:name="GearAppType"
        android:value="wgt" />


```

Obrázek 9 - Ukázka Android Manifestu

Zdroj: Vlastní

4.2.2 Singletony

Instance třídy `Connection`, která obstarává komunikaci s aplikací na hodinkách, se vytváří, podle nastavení v manifestu, když se hodinky připojují k mobilu. Inspiroval jsem se návrhovým vzorem Singletonu. Při vytváření instance třídy `Connection` je vytvořena instance třídy `SingletonConnection`, do které je uložena instance třídy `Connection`. Pomocí této implementace lze snadno přistupovat z aktivity hřiště k metodám třídy `Connection` a posílat informace do hodinek. Vytvořením Singletonu i pro aktivity hřiště jsem vyřešil další dva

problémy. Třída `Connection` potřebuje přístup k nejnovější aktivitě hřiště pro změnu jamky, a také pro zjištění, která aktivita hřiště smí odesílat data na hodinky. Protože v menu mobilní aplikace se při zmáčknutí tlačítka *Jdi na hřiště* vždy vytvoří nová aktivita hřiště a zde je důležité povolit odesílání dat jen té poslední.

4.2.3 Navázání spojení z pohledu aplikace na OS Android

Mobilní aplikace reaguje na žádosti o spojení od hodinek. Funkce z hodinek `SAAgent.findPeerAgents` posílá broadcastem *SAP agenta*, na toho odpoví metoda `onFindPeerAgentResponse`, pokud se nastavení *SAP agentů* bude shodovat. Když nebude shoda, tak parametr *result* metody `SAAgent.onFindPeerAgentResponse` se bude rovnat `null`. Dále android aplikace čeká. Pokud se hodinky rozhodnou o připojení, žádost je poslána funkcí `SAAgent.requestServiceConnection`. Tahle žádost je zpracována metodou `onServiceConnectionRequested`, která rozhodne o přijmutí žádosti. Výchozí implementace přijímá všechny žádosti. Jestliže se žádost přijme, hodinkám odpovídá metoda `onServiceConnectionResponse` a s odpovědí posílá `SASocket`.

4.2.4 Přijímání a odesílání dat

Informace z mobilu na hodinky se posílají ve třech případech. Když se vytvoří nová aktivita hřiště, nebo když dojde ke změně dat a při připojení hodinek. V aktivitě hřiště, v místech výpisu informací, byla přidána část kódu pro poslání těchto informací pomocí metody `sendNotification` třídy `Connection`. Metoda `sendNotification` umožňuje posílat proud dat jen té poslední aktivitě hřiště, která se používá.

Metoda `onReceive` zachycuje příchozí proud dat a pomocí metod tlačítek se posouvá jamka. Metody aktivity musí být spuštěny ve vláknu aktivity.

4.2.5 Odpojení

Android aplikace uzavírá spojení voláním metody `close` v objektu `SASocket`. Tím skončí spojení se vzdáleným zařízením, které je upozorněné metodou `onServiceConnectionLost`.

Pro uzavření spojení Gear web aplikace používá funkci `SASocket.close`.

5 INSTALAČNÍ PŘÍRUČKA

Soubory projektu jsou k dispozici v příloženém CD k této bakalářské práci. K spuštění této aplikace je zapotřebí fyzické zařízení:

- Mobilní telefon
- Chytré hodinky

Dále je potřeba stáhnout a nainstalovat vývojové prostředí pro OS Tizen. Teď bych doporučil nové Tizen Studio. Zde je již v instalačním balíčku rozšíření na mobily a hodinky. Dále je potřeba nainstalovat ovladač pro detekci Samsung zařízení vývojovým prostředím.

Ve vývojovém prostředí je nutné vytvořit certifikát autora a certifikát zařízení. Certifikát zařízení se nahraje přes Tizen Studio na hodinky.

Když je certifikát nahrán na hodinkách, tak již lze jednoduše nainstalovat aplikaci přes Tizen Studio.

Aplikaci na mobil můžeme nahrát přes datový kabel a jednoduše nainstalovat.

6 ZÁVĚR

Pro vývoj aplikace na chytré hodinky bylo vybráno vývojové prostředí Tizen SDK. Tizen SDK je oficiální vývojové prostředí pro Samsung Gear, takže její vývoj je podporován společností Samsung, a proto jsou také na stránkách Samsungu různé tutoriály a návody. Zároveň na stránkách Tizen jsou další návody, články o rozhraní, a jiné. Takže uživatel má hodně materiálů a podkladů pro porozumění vývoji v Tizen SDK.

Druhá aplikace, aplikace na mobilním telefonu, je původně napsaná ve vývojovém prostředí Android Studio. Z tohoto důvodu a s doporučením jsem i já upravoval tuto aplikaci v Android Studiu. Práce probíhala rychle, Android Studio je intuitivní, přehledné a velice dobré vývojové prostředí s našeptávačem. Ale jelikož jsem se s ním dříve neseťkal, tak jsem měl problémy s buildovacím systémem Gradle.

Teoretická část práce se zaměřuje na okolnosti chytrých hodinek a mobilů. Jsou představeny operační systémy mobilních zařízení, je zde i popis některých programovacích jazyků v této oblasti. Dále se teoretická část zabývá technologiemi použité při vývoji aplikace.

V praktické části jsou představeny hlavní cíle a požadavky na mobilní aplikaci a popsán způsob implementace, který zaručuje splnění zadaných požadavků. Při vývoji aplikace došlo k rozšíření původních plánů, hlavně v oblasti distribuce. Z důvodů bezpečnostních prvků, při instalaci aplikace na chytré hodinky, je distribuce této aplikace, bez přidání aplikace do obchodu Samsung, velice obtížná. Aplikace již byla vystavená na Samsung Store a i v Google Store je již upravená verze.

Po dokončení všech náležitostí je naplánované upravení aplikace i na kulaté displeje chytrých hodinek neboli Gear S2 a Gear S3, aby i ostatní potenciální uživatelé mohli využít aplikaci. Předpokládá se, že časem bude i zpětná vazba a případné rozšíření i na ostatní golfové hřiště. Aplikace je bez reklam a zdarma ke stažení.

7 POUŽITÁ LITERATURA

1. MOZILLA DEVELOPER NETWORK AND INDIVIDUAL CONTRIBUTORS. A re-introduction to JavaScript. Mozilla *Developer Network* [online]. 2015 [cit. 25.05.2016]. Dostupné z: https://developer.mozilla.org/en-US/docs/Web/JavaScript/A_re-introduction_to_JavaScript
2. **JACOBSEN, Joachim Cohn** a spol. *What is CSS?* [online]. 2015 [cit. 29.05.2016]. Dostupné z: <http://www.CSS.net/tutorials/css/lesson1.php>
3. **KOCHAN, Stephen G.** *Objective-C 2.0: výukový kurz programování pro Mac OS X a iPhone*. Brno: Computer Press, 2010. ISBN 978-80-251-2654-7.
4. **REIFMAN, Jeff.** *What is HTML5?* [online]. 17.03.2016 [cit. 15.06.2016]. Dostupné z: <https://code.tutsplus.com/tutorials/what-is-html5--cms-25803>
5. SAMSUNG DEVELOPER. *Creating your first app* [online]. 2010 [cit. 15.03.2017]. Dostupné z: <http://developer.samsung.com/gear/develop/creating-your-first-app>
6. TIZEN PROJECT. *Introduction to Web Applications* [online]. 2012 [cit. 17.03.2017]. Dostupné z: <https://developer.tizen.org/ko/development/training/web-application>
7. **RAO, Raja.** *WearablePack-SamsungGear2* [online]. 2014 [cit. 04.05.2016]. Dostupné z: <https://github.com/developforce/WearablePack-SamsungGear2/tree/master/samples/TodayApp>
8. TIZEN PROJECT. *Issuing a Tizen Certificate* [online]. 2012 [cit. 18.02.2016]. Dostupné z: <https://developer.tizen.org/ko/community/tip-tech/issuing-tizen-certificate-certificate-extension-ver-1.1.1?langredirect=1>
9. **NADOBA, Adam.** Developing your first Tizen Galaxy Gear2 App – Tips In: *Scalac* [online]. 30.07.2014 [cit. 04.06.2016]. Dostupné z: <https://blog.scalac.io/2014/07/30/developing-your-first-galaxy-gear-app.html>
10. *Update the IDE and Tools*. In: *Android Studio* [online]. Creative Commons, 2016 [cit. 04.05.2016]. Dostupné z: <https://developer.android.com/studio/intro/update.html>
11. *Create and Manage Virtual Devices*. In: *Android Studio* [online]. Creative Commons, 2016 [cit. 04.05.2016]. Dostupné z: <https://developer.android.com/studio/run/managing-avds.html>
12. TIZEN PROJECT. *Overview* [online]. 2012 [cit. 18.02.2016]. Dostupné z: <https://developer.tizen.org/ko/development/tizen-studio?langredirect=1>
13. **POTHITOS, Adam.** *Mobile Industry Review* [online]. 31.10.2016 [cit. 28.08.2017]. Dostupné z: <https://www.mobileindustryreview.com/2016/10/the-history-of-the-smartphone.html>
14. *iOS* [online]. APPLE INC. 2016 [cit. 08.05.2016]. Dostupné z: <http://www.apple.com/cz/ios/>

15. **HAMAL, Tomáš.** *Návrh a implementace aplikace GPS golfového dálkoměru pro mobilní telefon s operačním systémem Android.* 2016. Bakalářská práce. Pardubice: Univerzita Pardubice, Fakulta informačních technologií. Vedoucí práce Fikejz, Jan
16. Android is for everyone. GOOGLE. *Opportunity. Powered by choice.* [online]. 2016 [cit. 2017-08-29]. Dostupné z: <https://www.android.com/everyone/enabling-opportunity/>
17. **BEAL, Vangie.** *Java* [online]. 2016 [cit. 29.08.2017]. Dostupné z: <http://www.webopedia.com/TERM/J/Java.html>
18. **HASAN, Noorul.** *History of C# Programming* [online]. 27.09.2012 [cit. 29.08.2017]. Dostupné z: <http://aboutsharpprogramming.blogspot.cz/2012/09/history-of-c-programming.html>
19. **JONES, Bradley L.** *What is C#?* [online]. Sams Publisher, 12.11.2001 [cit. 29.08.2017]. Dostupné z: <http://www.developer.com/net/asp/article.php/922211/What-is-C.htm>
20. **KARÁSEK, Jakub.** *Recenze Windows 10 Mobile: Budoucnost je univerzální* [online]. 04.01.2016 [cit. 25.05.2016]. ISSN 1801-3066. Dostupné z: <https://smartmania.cz/recenze-windows-10-mobile-microsoft-test-12777/>

8 PŘÍLOHY

Příloha A Diagram	39
-------------------------	----

Příloha A Diagram

